Scheduling Constrained Dynamic Applications on Clusters

Arun Chauhan

joint work with

Kathleen Knobe

Jim Rehg

Nikhil Rishiyur

Umakishore Ramachandran

November 17

SC 1999, Portland, OR

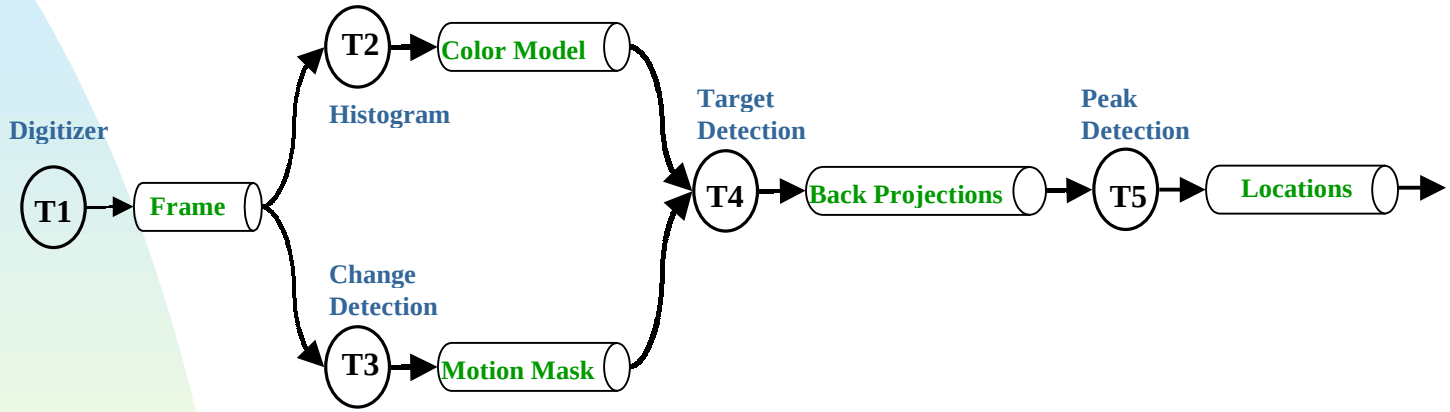
Context: user's view

- Free Standing Smart Kiosk
- Automatically detects approaching customers
- Animated face exhibits natural gaze behavior
- Interacts through synthesized voice and touch-screen

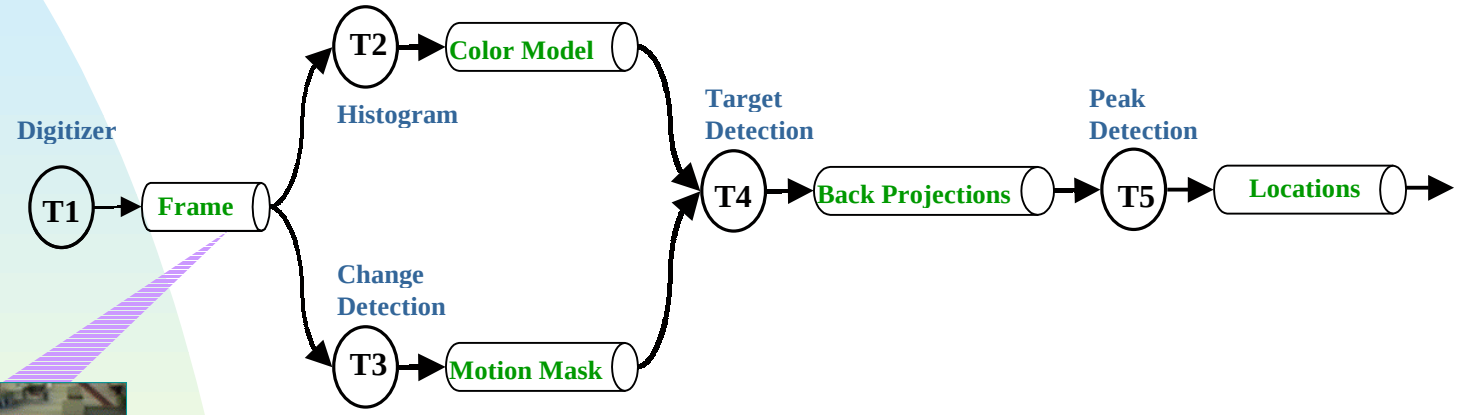
Context: programmer's view

- Multi-media applications
  - ⇒ streaming data
- Interactive
  - ⇒ response time (latency)
- Needs to be compelling
  - ⇒ natural gaze behavior (people tracking)
- Kiosk has other background apps
  - ⇒ dynamic environment

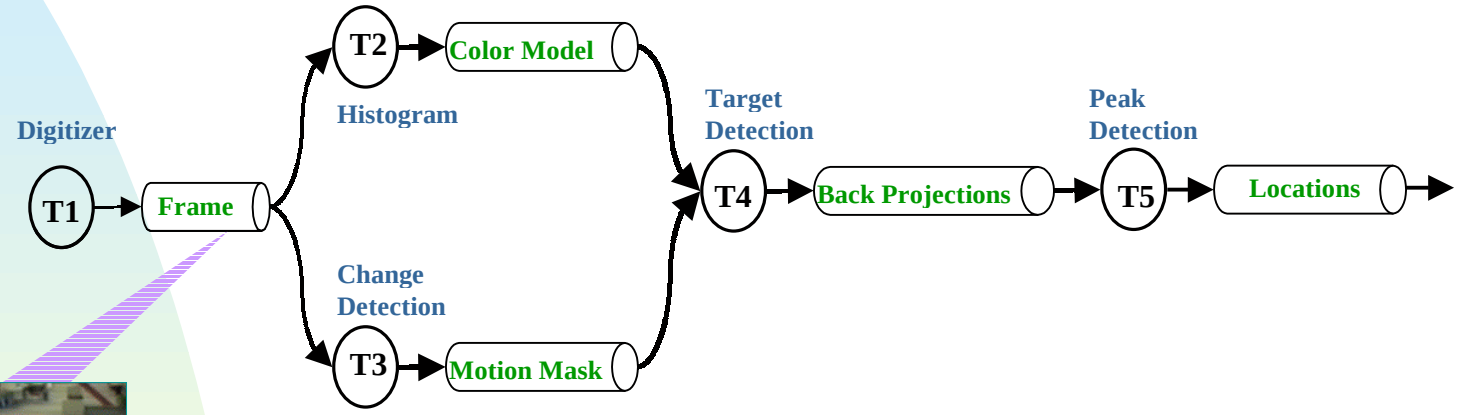
Color Tracker



Color Tracker



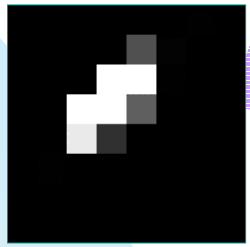
Color Tracker

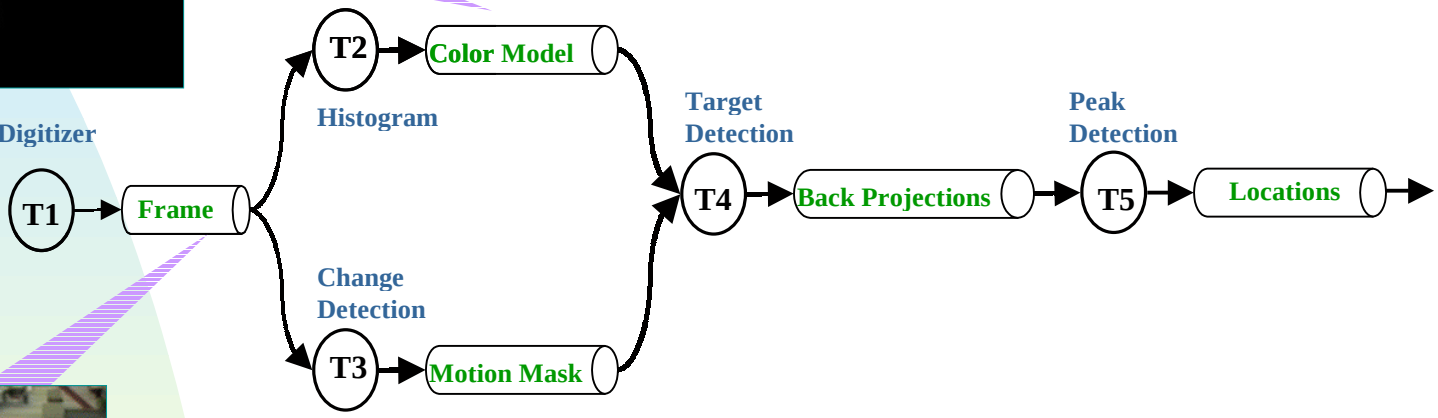


Model 1

Model 2

Color Tracker

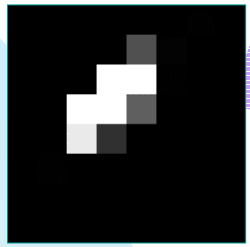


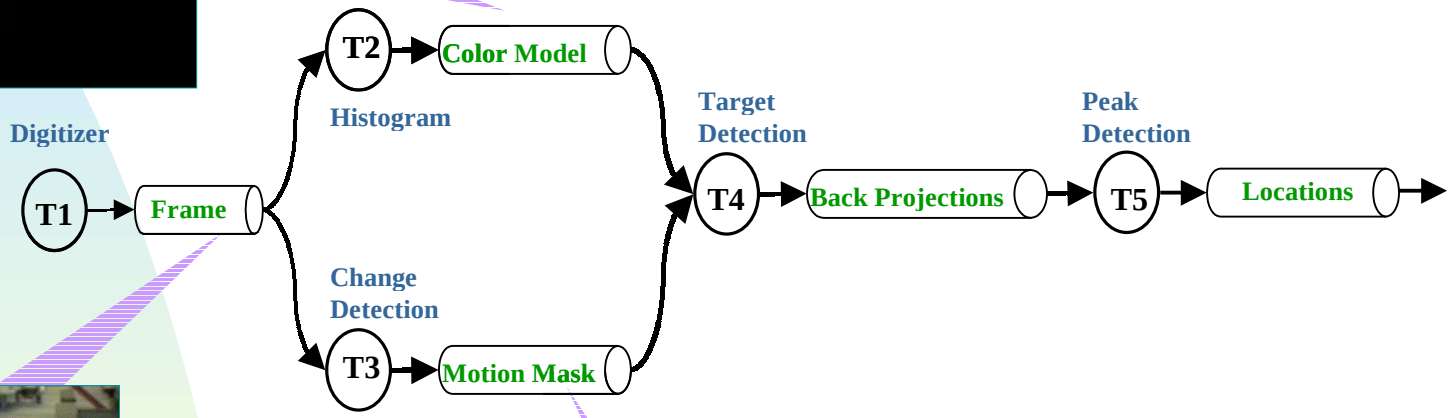


Model 1

Model 2

Color Tracker

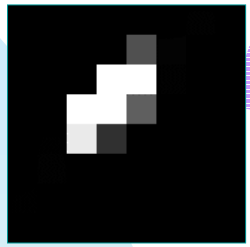


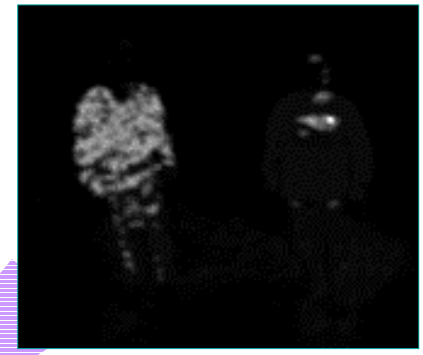


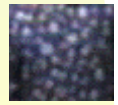
Model 1

Model 2

Color Tracker

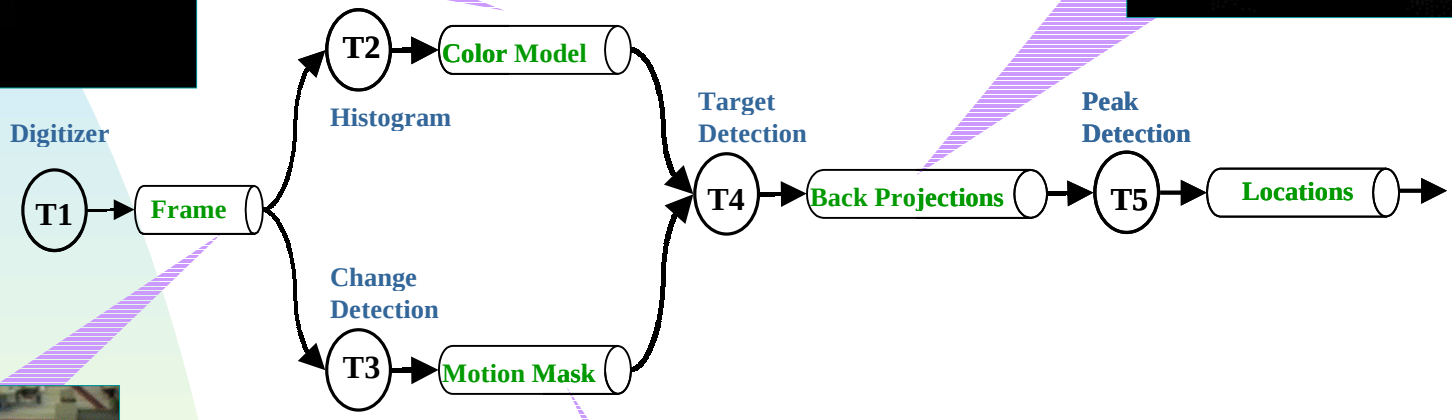




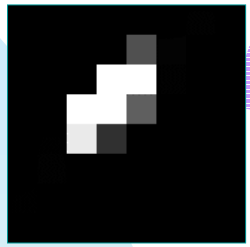


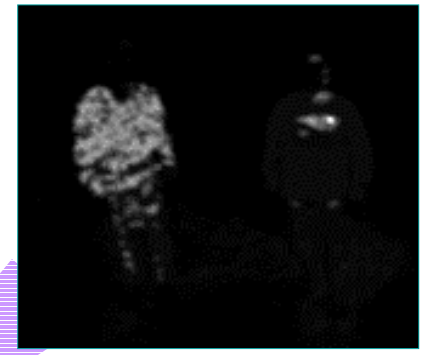
Model 1

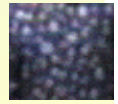
Model 2



Color Tracker

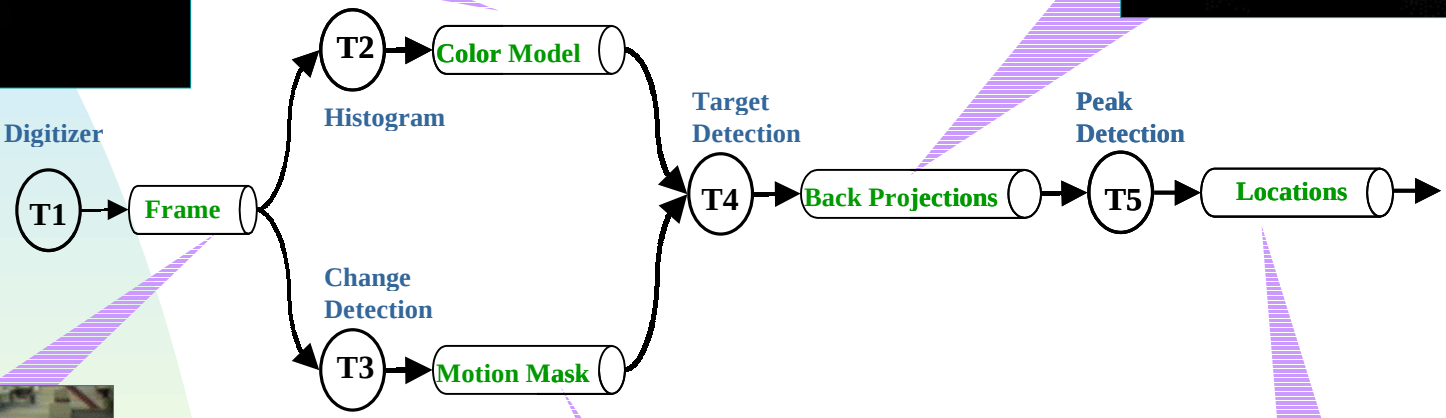




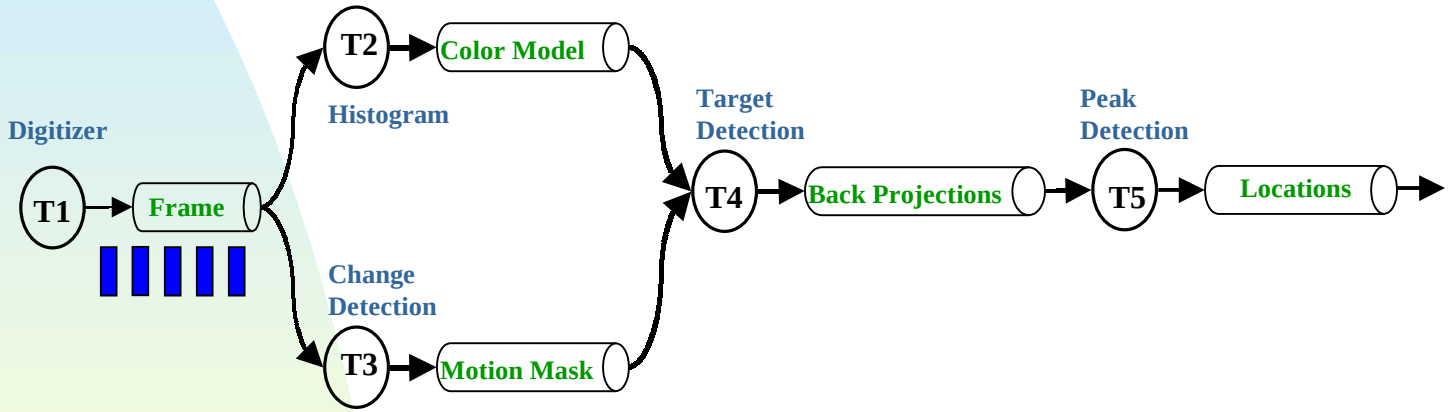


Model 1

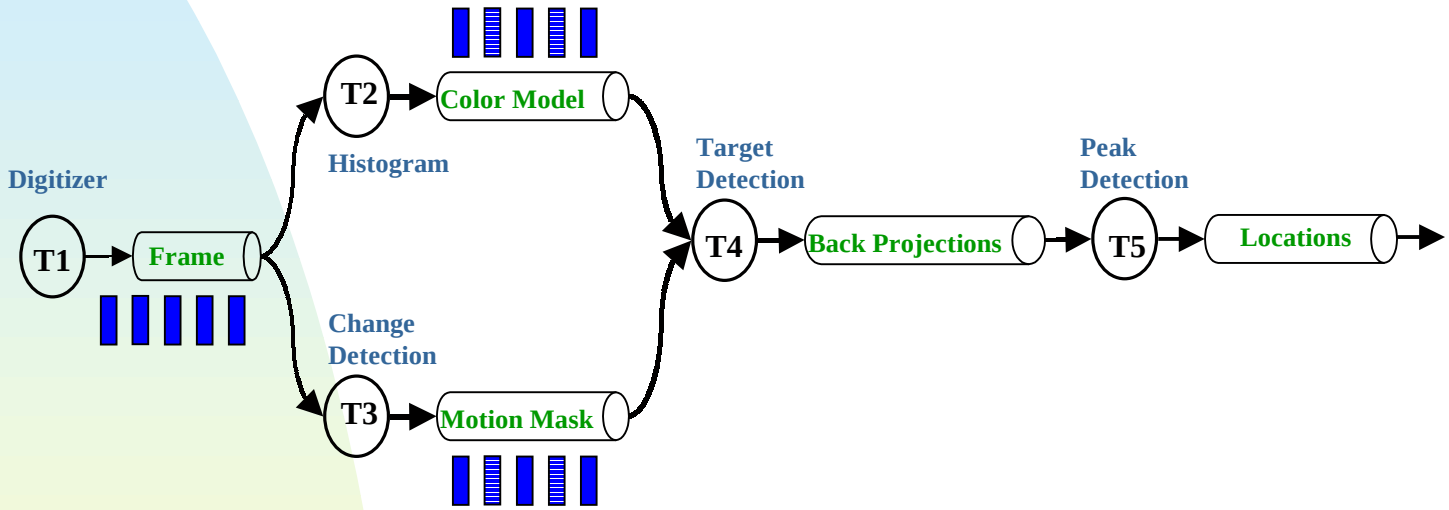
Model 2



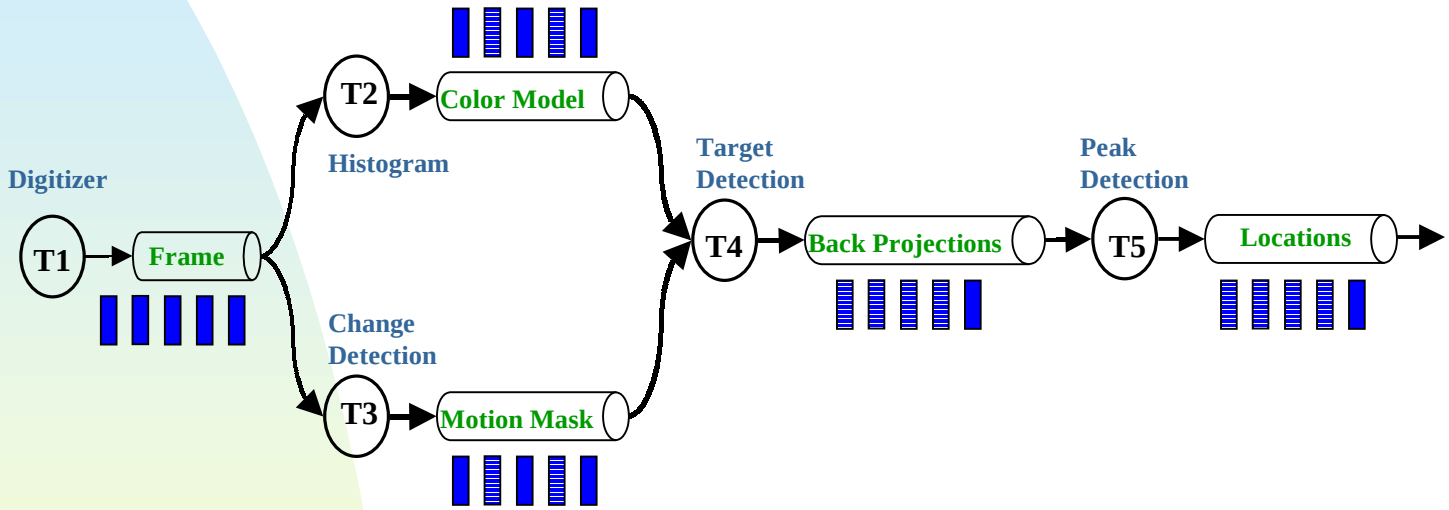
Closer Look at Tracker



Closer Look at Tracker



Closer Look at Tracker



Characteristics

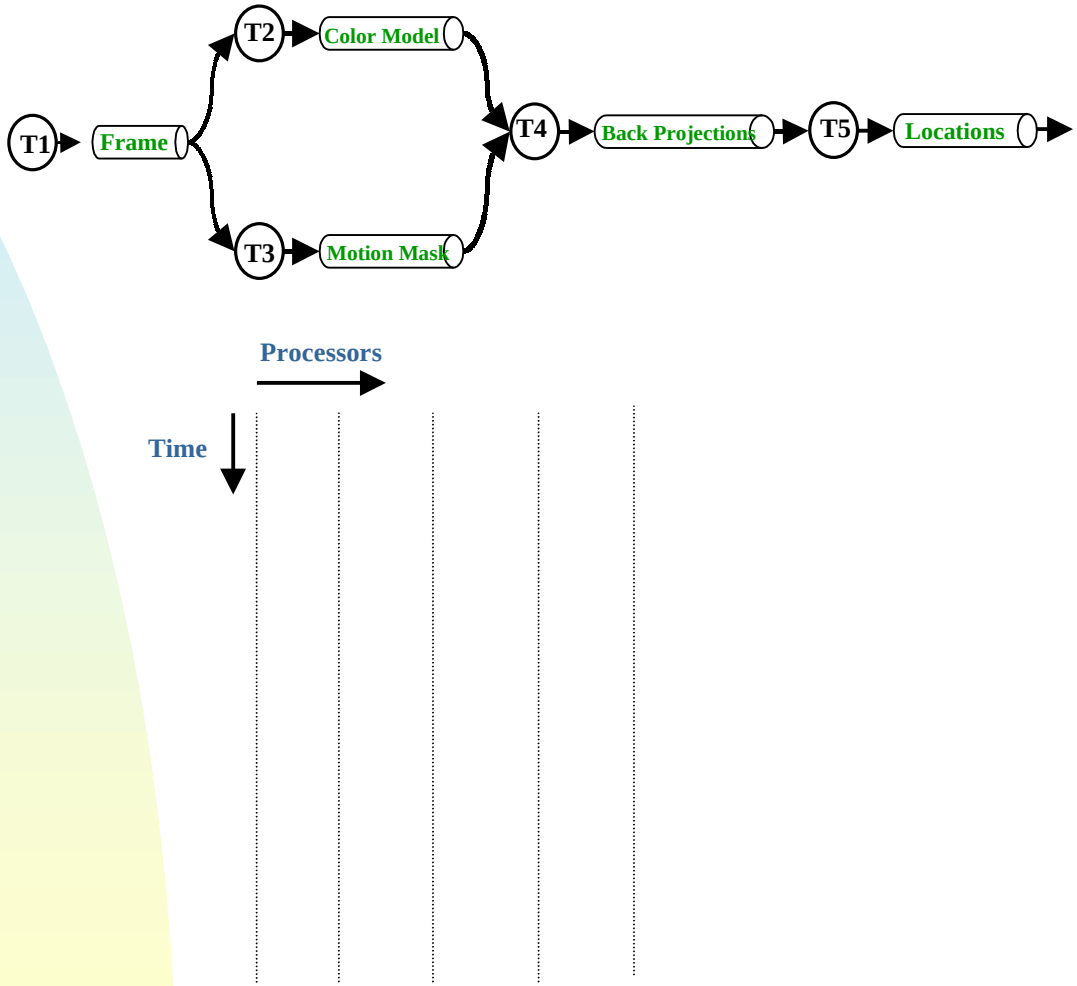
- Downstream tasks are more compute intensive
- Tasks “sample” the stream at varying rates
- Fundamental ability to sample at varying rates provided by Space-Time Memory

Characteristics

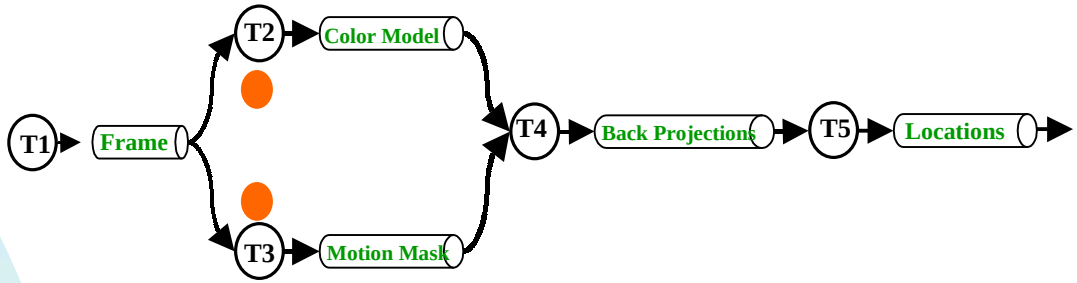
- Downstream tasks are more compute intensive
- Tasks “sample” the stream at varying rates
- Fundamental ability to sample at varying rates provided by Space-Time Memory

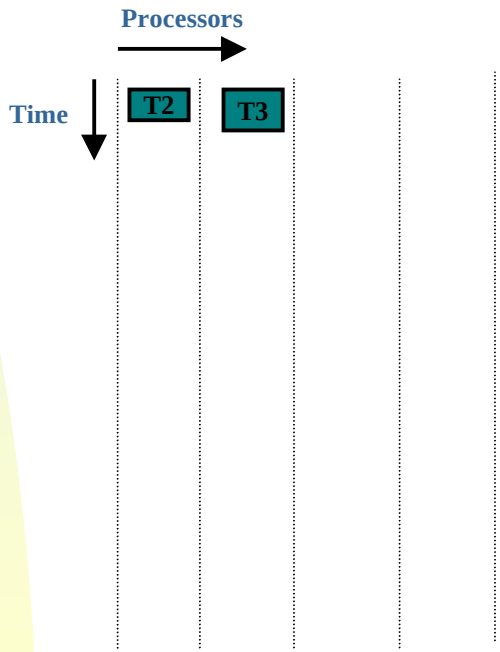
Raises scheduling questions

The Problem

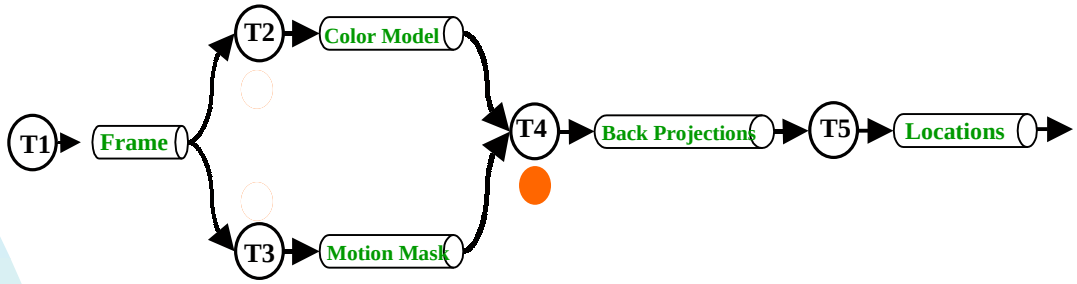

T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations
Processors
Time

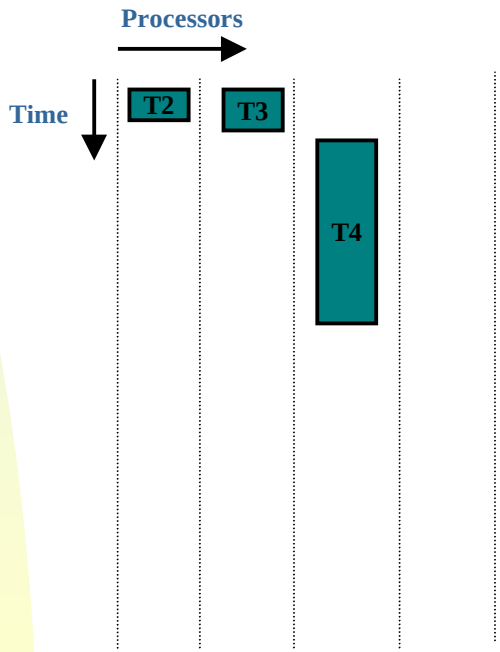
The Problem


T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations

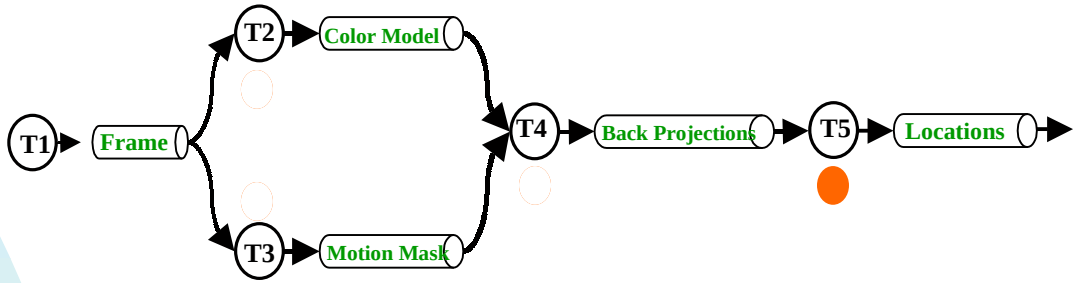

Processors
Time
T2
T3

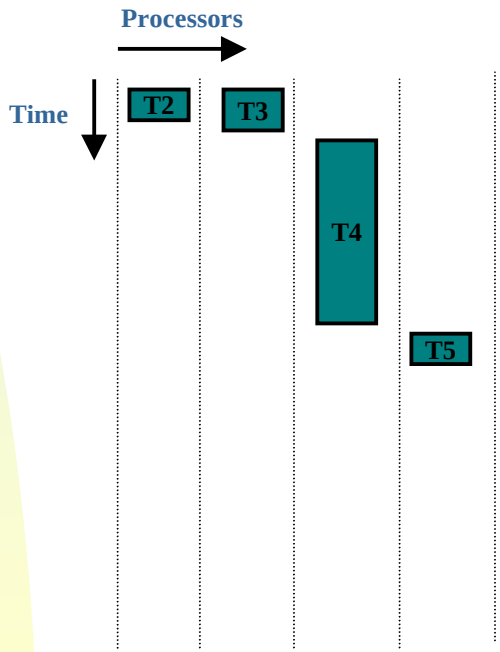
The Problem


T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations

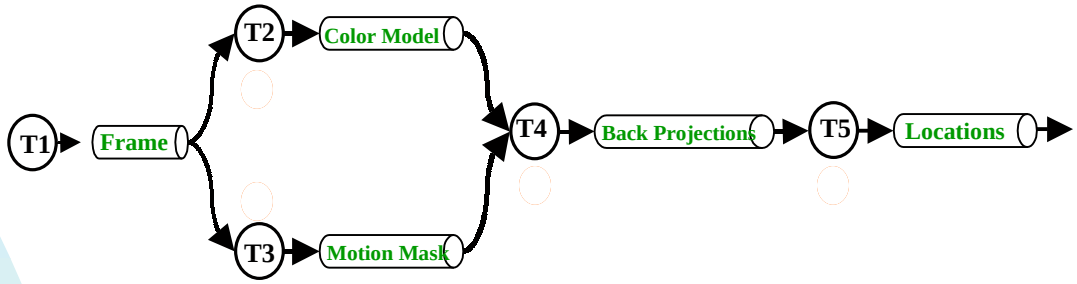

Processors
Time
T2
T3
T4

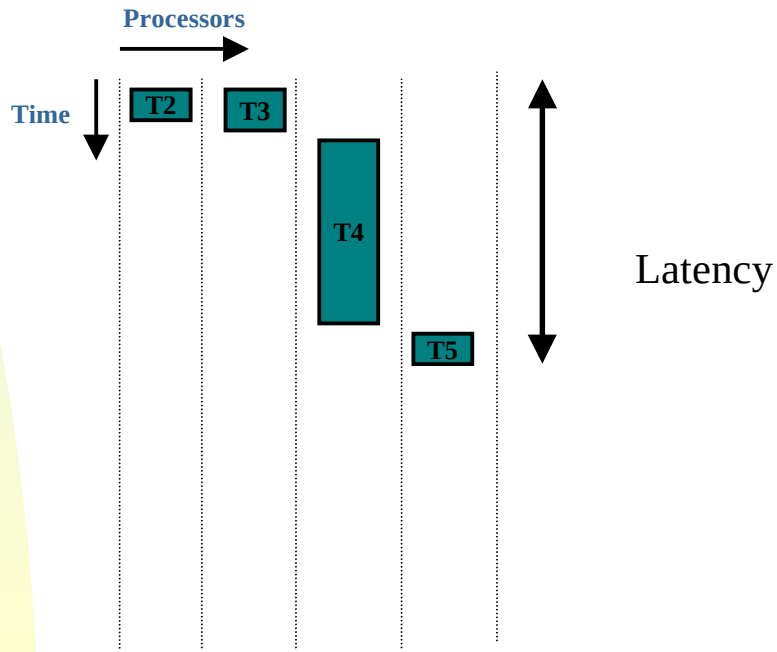
The Problem


T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations

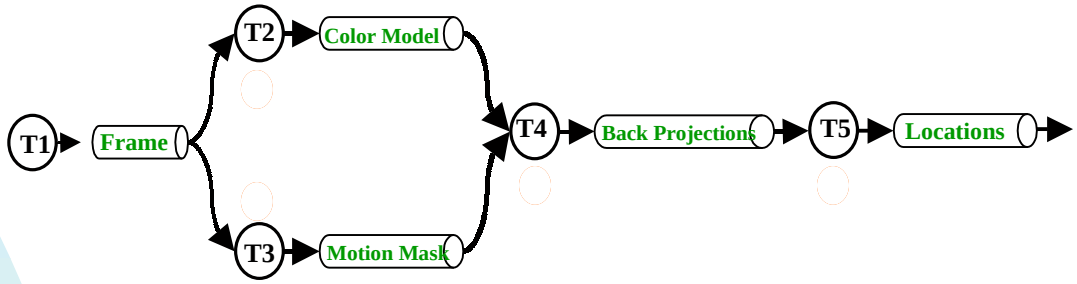

Processors
Time
T2
T3
T4
T5

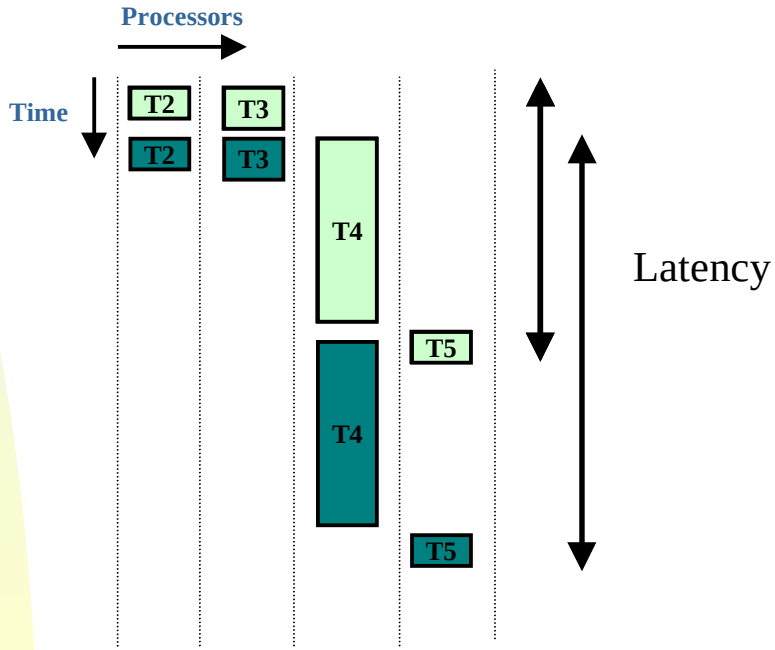
The Problem


T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations


Processors
Time
T2
T3
T4
T5
Latency

The Problem


T1
Frame
T2
Color Model
T3
Motion Mask
T4
Back Projections
T5
Locations


Processors
Time
T2
T2
T3
T3
T4
T4
T5
T5
Latency

Metrics

- Good use of resources
  - good throughput
- Kiosk must be compelling and interactive
  - low latency per frame
  - avoid “dead” periods
- Use general-purpose (“commodity”) OS and hardware

Metrics

- Good use of resources
  - good throughput
- Kiosk must be compelling and interactive
  - low latency per frame
  - avoid “dead” periods
- Use general-purpose (“commodity”) OS and hardware

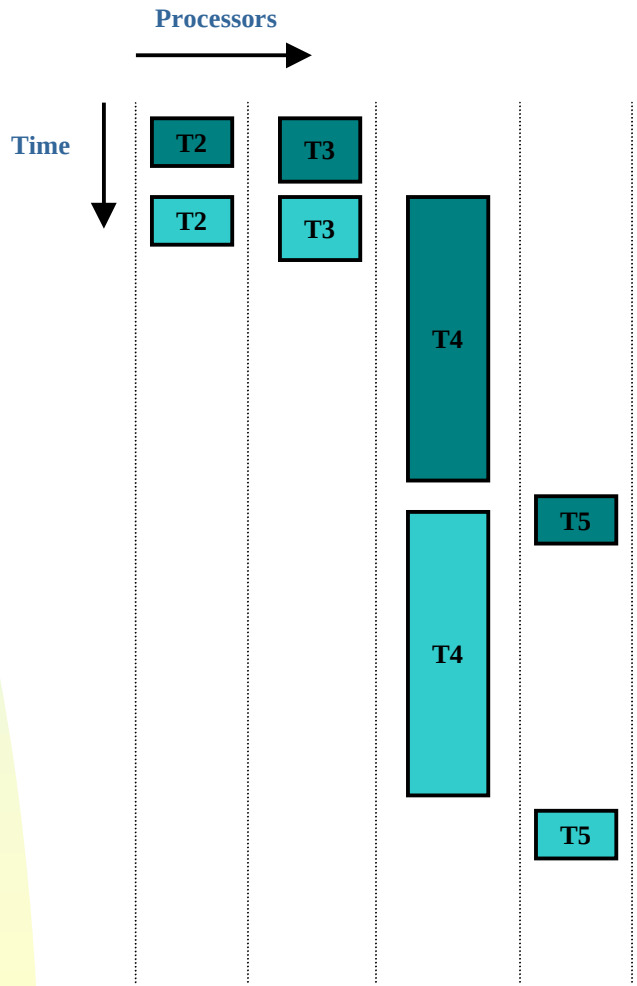
Do this in a dynamic environment

Constrained Dynamism

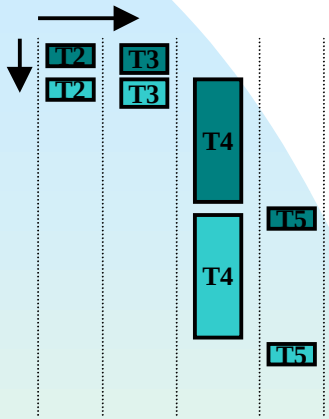
- The system changes among a small number of states
  - run-time environment, e.g., number of processors available, or load
  - input dependent, e.g., number of users
- State changes are infrequent
- State changes are detectable

Improving the Schedules

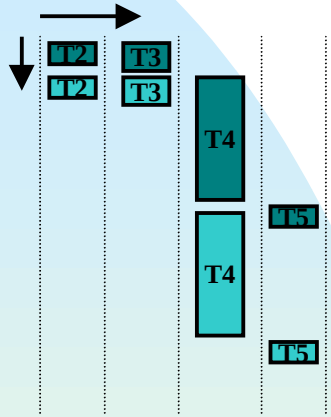
Improving the Schedules

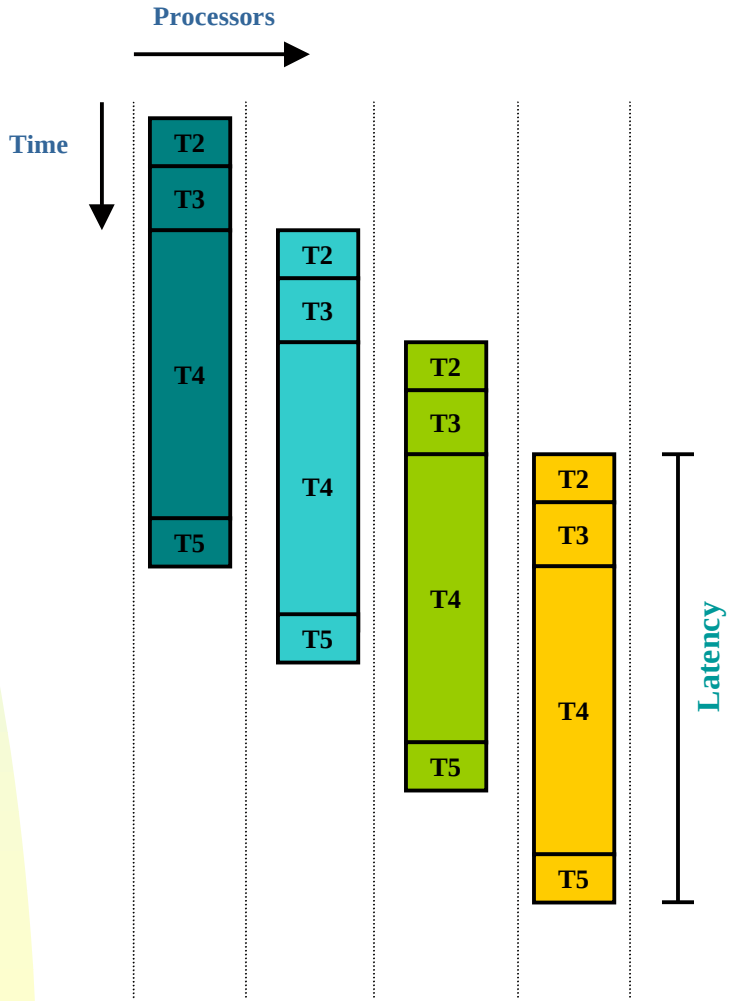


Improving the Schedules

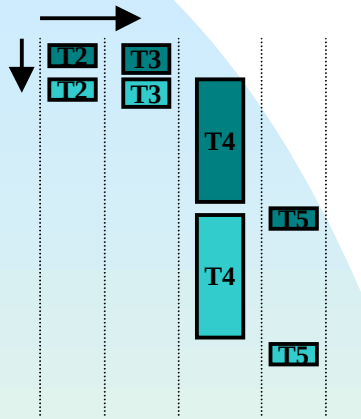


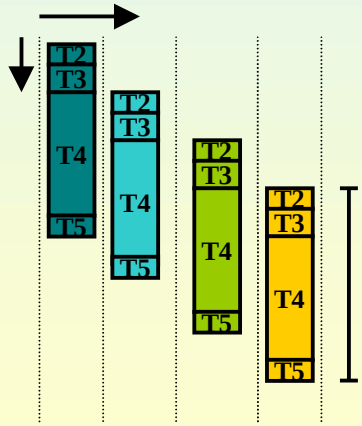
Improving the Schedules



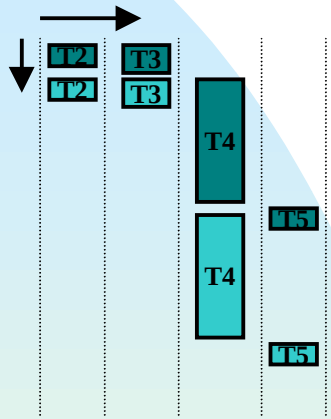


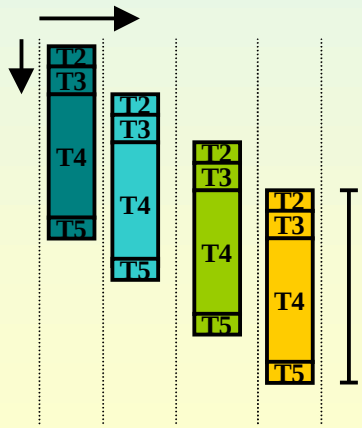
Improving the Schedules

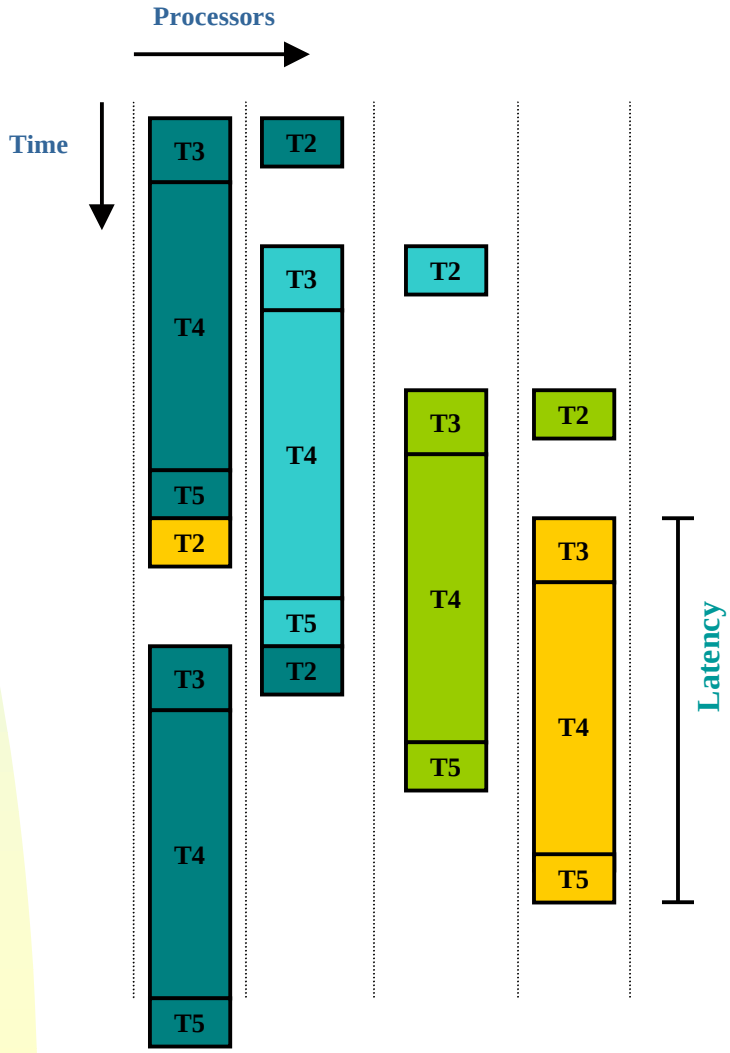




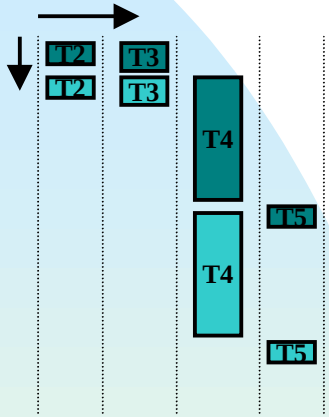
Improving the Schedules

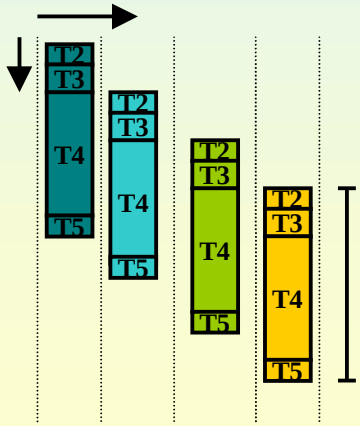


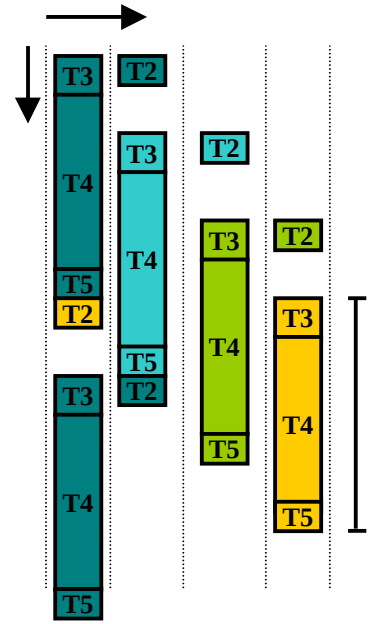




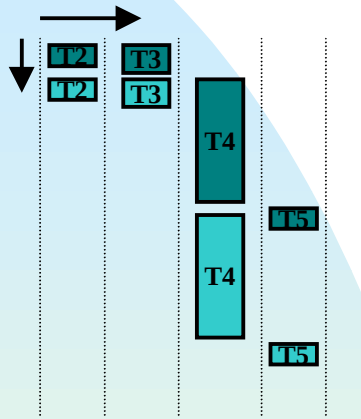
Improving the Schedules

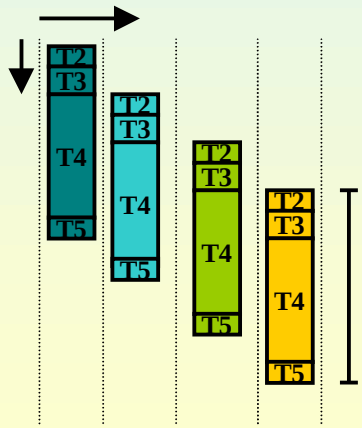


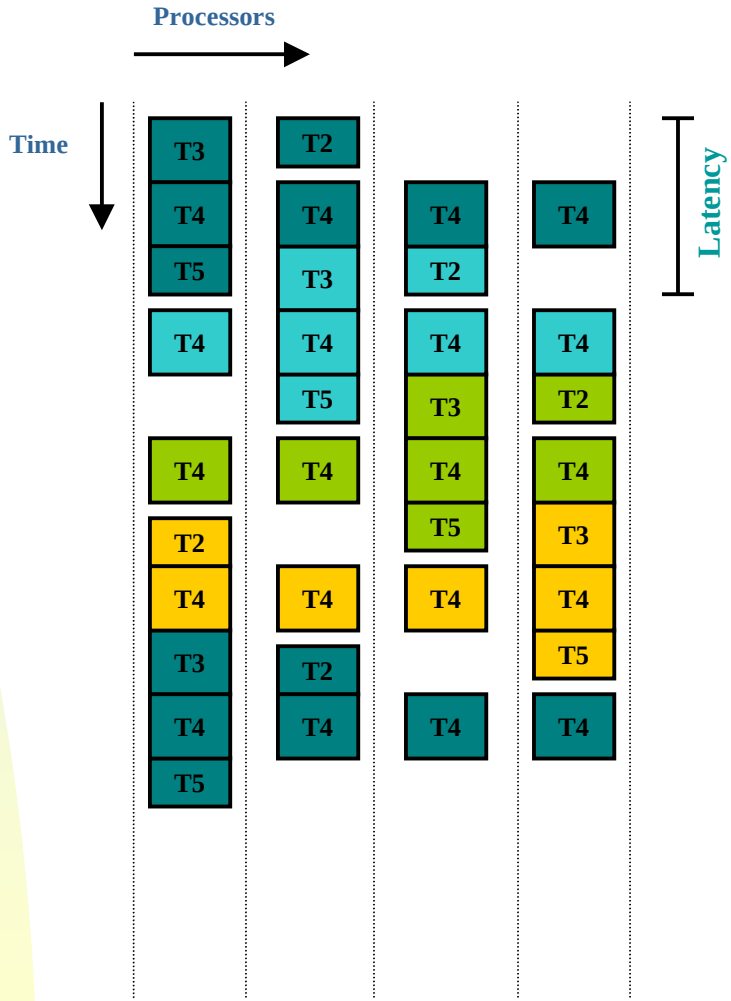


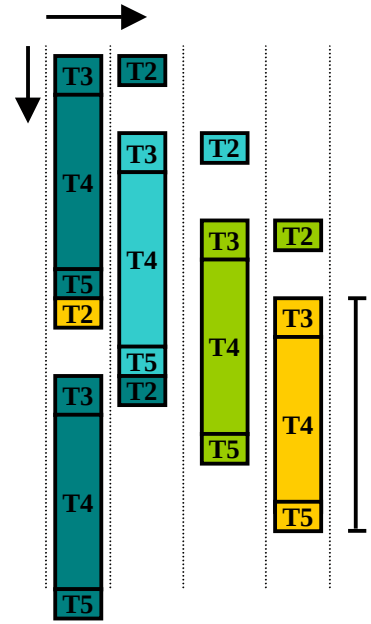


Improving the Schedules







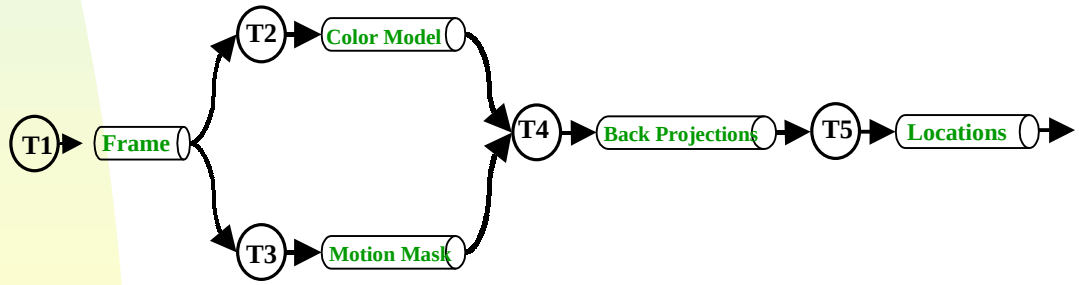


How does this work?

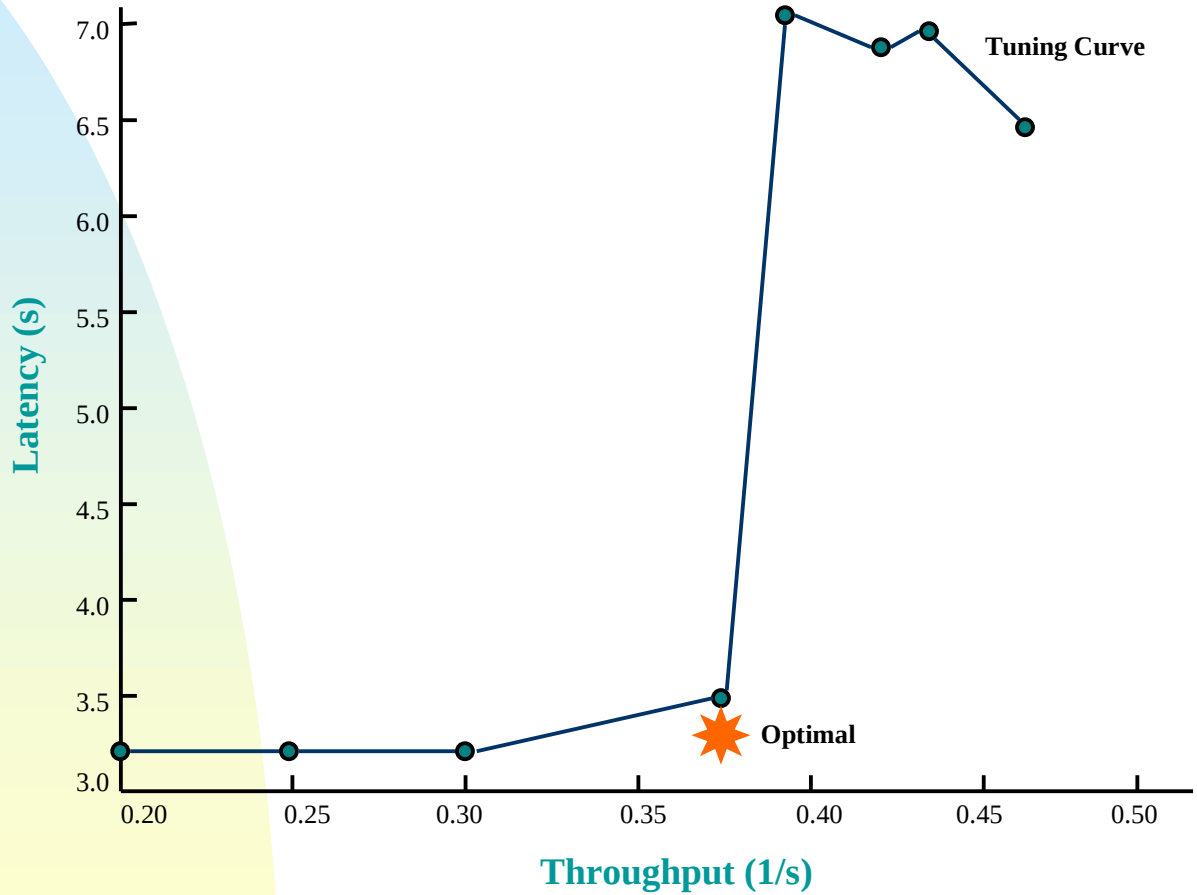
- Compute optimal schedules for each state
  - input: execution time, communication time
  - compute: minimal latency, single iteration schedule for minimal latency, and finally multiple iteration schedule
- Detect the current state at run-time and choose the best schedule

Why does this work?

- Alternatives
  - Do nothing!
  - Control rate of frame generation
  - Control the size of inter-task “channels”
- A limited number of states is the key



Comparison


7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
Latency (s)
0.20
0.25
0.30
0.35
0.40
0.45
0.50
Throughput (1/s)
Tuning Curve
Optimal

Benefits

- No extra work (good resource utilization)
- Reduces “live” time (smaller space req.)
- Simplifies garbage collection
- Implicitly solves flow control

Benefits

- No extra work (good resource utilization)
- Reduces “live” time (smaller space req.)
- Simplifies garbage collection
- Implicitly solves flow control

Works on commodity systems

Data Decomposition

	Total Models		
	1	8	
Partitions	MP=1	MP=8	MP=1
FP=1	0.876	1.857	6.850
FP=4	0.275	2.155	2.033

- Number of input models defines a state
- Small number of states $\Rightarrow$ constrained dynamism

Conclusion

- A class of applications exhibits the property of “constrained dynamism”
- The property enables state-based approach to obtain good schedules in the face of dynamic environment
- Constrained Dynamism also helps in other aspects of application tuning, like, parallelization strategy